

Consultative Committee for Space Data Systems

**DRAFT RECOMMENDATION FOR SPACE
DATA SYSTEM STANDARDS**

DATA ENTITY DICTIONARY SPECIFICATION LANGUAGE (DEDSL) (CCSD0011)

CCSDS 647.0-R-2.2

DRAFT RED BOOK

January 1999



AUTHORITY

Issue:	Red Book, Issue 2.2
Date:	January 1999
Location:	

**(WHEN APPROVED FOR PUBLICATION AS A RECOMMENDATION, THE
FOLLOWING TEXT WILL APPEAR)**

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and represents the consensus technical agreement of the participating CCSDS Member Agencies. The procedure for review and authorisation of CCSDS Recommendations is detailed in Reference [E1], and the record of Agency participation in the authorisation of this document can be obtained from the CCSDS Secretariat at the address below.

This Recommendation is published and maintained by:

CCSDS Secretariat
Program Integration Division (Code-OI)
National Aeronautics and Space Administration
Washington, DC 20546, USA

STATEMENT OF INTENT

(WHEN APPROVED FOR PUBLICATION AS A RECOMMENDATION, THE FOLLOWING TEXT WILL APPEAR)

The Consultative Committee for Space Data Systems (CCSDS) is an organisation officially established by the management of the member space Agencies. The committee meets periodically to address data system problems that are common to all participants and to formulate sound technical solutions to these problems. Inasmuch as participation in the CCSDS is completely voluntary, the results of the committee are termed **Recommendations** and are not considered binding on any Agency.

This **Recommendation** is issued by, and represents the consensus of, the CCSDS Plenary body. Agency endorsement of **Recommendations** is entirely voluntary. Endorsement, however, indicates the following understandings:

- Whenever an Agency establishes a CCSDS-related **standard**, this **standard** will be in accordance with the relevant **Recommendation**. Establishing such a **standard** does not preclude other provisions which an Agency may develop.
- Whenever an Agency establishes a CCSDS-related **standard**, the Agency will provide other CCSDS member Agencies with the following information:
 - The **standard** itself.
 - The anticipated date of initial operational capability.
 - The anticipated duration of operational service.
- Specific service arrangements shall be made via memorandum of agreement. Neither this **Recommendation** nor any ensuing **standard** is a substitute for a memorandum of agreement.

No later than five years from its date of issuance, this **Recommendation** will be reviewed by the CCSDS to determine whether it should: (1) remain in effect without change; (2) be changed to reflect the impact of new technologies, new requirements, or new directions; or (3) be retired or cancelled.

In those instances when a new version of a **Recommendation** is issued, existing CCSDS-related Agency **standards** and implementation are not negated or deemed to be non-CCSDS compliant. It is the responsibility of each Agency to determine when such **standards** or implementation are to be modified. Each Agency is, however, strongly encouraged to direct planning of its new **standards** and implementations towards the later versions of this **Recommendation**.

FOREWORD

This Recommendation is a technical Recommendation that provides a model and language to increase the standardisation of the expression of semantic concepts that are to be carried with data. These semantic concepts are given standard names and a standard way of expressing them is also provided. The semantic information may be conveyed either in a computer processable manner or via conventional (e.g. paper) documentation.

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This document is therefore subject to CCSDS document management and change control procedures which are defined in Reference [E1].

Questions relating to the contents or status of this document should be addressed to the CCSDS Secretariat at the address indicated on page I.

At time of publication, the active Member and Observer Agencies of the CCSDS were:

Member Agencies

- British National Space Centre (BNSC)/United Kingdom.
- Canadian Space Agency (CSA)/Canada.
- Centre National d'Etudes Spatiales (CNES)/France.
- Deutsche Forschungsanstalt für Luft- und Raumfahrt e.V. (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- National Aeronautics and Space Administration (NASA)/USA.
- National Space Development Agency of Japan (NASDA)/Japan.
- Russian Space Agency (RSA)/Russian Federation.

Observer Agencies

- Australian Space Office (ASO)/Australia.
- Austrian Space Agency (ASA)/Austria.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- Centro Tecnico Aeroespacial (CTA)/Brazil.
- Chinese Academy of Space Technology (CAST)/China.
- Communications Research Laboratory (CRL)/Japan.
- Danish Space Research Institute (DSRI)/Denmark.
- European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
- European Telecommunications Satellite Organization (EUTELSAT)/Europe.
- Federal Service of Scientific, Technical & Cultural Affairs (FSST&CA)/Belgium.
- Hellenic National Space Committee (HNSC)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Industry Canada/Communications Research Centre (CRC)/Canada.
- Institute of Space and Astronautical Science (ISAS)/Japan.
- Institute of Space Research (IKI)/Russian Federation.
- KFKI Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- MIKOMTEK: CSIR (CSIR)/Republic of South Africa.
- Korea Aerospace Research Institute (KARI)/South Korea.
- Ministry of Communications (MOC)/Israel.
- National Oceanic & Atmospheric Administration (NOAA)/USA.
- National Space Program Office (NPSO)/Taiwan.
- Swedish Space Corporation (SSC)/Sweden.
- United States Geological Survey (USGS)/USA.

DOCUMENT CONTROL

Document	Title	Date	Status/Remarks
CCSDS 647.0-R-1.0	Recommendation for Space Data System Standards: Data Entity Dictionary Specification Language (DEDSL)	November 1996	Original Issue
CCSDS 647.0-R-2.0	Recommendation for Space Data System Standards: Data Entity Dictionary Specification Language (DEDSL)	March 1998	New Issue
CCSDS 647.0-R-2.1	Recommendation for Space Data System Standards: Data Entity Dictionary Specification Language (DEDSL)	September 1998	Updated
CCSDS 647.0-R-2.2	Recommendation for Space Data System Standards: Data Entity Dictionary Specification Language (DEDSL)	January 1999	Updated

CONTENTS

<u>Section</u>	<u>Page</u>
1 INTRODUCTION	1-1
1.1 PURPOSE AND SCOPE.....	1-1
1.2 APPLICABILITY	1-1
1.3 RATIONALE	1-2
1.4 DOCUMENT STRUCTURE	1-2
1.5 DEFINITIONS	1-3
1.5.1 ACRONYMS AND ABBREVIATIONS.....	1-3
1.5.2 GLOSSARY OF TERMS.....	1-4
1.5.3 NOMENCLATURE.....	1-7
1.6 REFERENCES.....	1-7
2 OVERVIEW	2-1
2.1 USES OF DATA ENTITY DICTIONARIES	2-1
2.1.1 PRODUCT DATA ENTITY DICTIONARIES	2-1
2.1.2 COMMUNITY DATA ENTITY DICTIONARIES.....	2-3
2.2 APPLICATION OF THE DEDSL	2-4
2.3 REGISTERING DED	2-5
3 ABSTRACT STANDARD.....	3-1
3.1 DESCRIPTORS OF A DATA ENTITY ATTRIBUTE	3-1
3.1.1 ATTRIBUTE_NAME.....	3-2
3.1.2 ATTRIBUTE_DEFINITION	3-3
3.1.3 ATTRIBUTE_OBLIGATION	3-3
3.1.4 ATTRIBUTE_CONDITION	3-4
3.1.5 ATTRIBUTE_MAXIMUM_OCCURRENCE	3-4
3.1.6 ATTRIBUTE_VALUE_TYPE	3-4
3.1.7 ATTRIBUTE_MAXIMUM_SIZE.....	3-5
3.1.8 ATTRIBUTE_ENUMERATION_VALUES	3-6
3.1.9 ATTRIBUTE_COMMENT	3-6
3.1.10 ATTRIBUTE_INHERITANCE.....	3-6
3.1.11 ATTRIBUTE_DEFAULT_VALUE	3-7
3.1.12 ATTRIBUTE_VALUE_EXAMPLE	3-7
3.2 DATA ENTITY ATTRIBUTES AND DICTIONARY ATTRIBUTES	3-8

3.2.1	CONCEPT DEFINITIONS.....	3-8
3.2.2	GENERAL VIEW OF THE STANDARD ATTRIBUTES	3-9
3.2.2.1	STANDARD ATTRIBUTES FOR DEDSL DICTIONARIES	3-9
3.2.2.2	STANDARD ATTRIBUTES FOR DATA ENTITIES.....	3-10
3.2.3	DEDSL DICTIONARY ATTRIBUTES	3-11
3.2.3.1	DICTIONARY_NAME.....	3-11
3.2.3.2	DICTIONARY_DEFINITION.....	3-11
3.2.3.3	EXTERNAL_DICTIONARY_REFERENCE.....	3-12
3.2.3.4	TEXT_FIELD_CHARACTER_SET	3-12
3.2.3.5	CASE_SENSITIVITY	3-12
3.2.3.6	DICTIONARY_LANGUAGE	3-13
3.2.3.7	DICTIONARY_VERSION	3-13
3.2.3.8	DICTIONARY_ADID	3-14
3.2.3.9	DEDSL_VERSION	3-14
3.2.4	DEDSL STANDARD ATTRIBUTES FOR DATA ENTITIES.....	3-15
3.2.4.1	IDENTIFYING ATTRIBUTES	3-15
3.2.4.1.1	NAME.....	3-15
3.2.4.1.2	ALIAS	3-15
3.2.4.1.3	CLASS	3-16
3.2.4.2	DEFINITIONAL ATTRIBUTES.....	3-17
3.2.4.2.1	DEFINITION	3-17
3.2.4.2.2	SHORT_DEFINITION	3-17
3.2.4.2.3	COMMENT	3-17
3.2.4.2.4	UNITS.....	3-18
3.2.4.2.5	SPECIFIC_INSTANCE	3-19
3.2.4.3	RELATIONAL ATTRIBUTES.....	3-19
3.2.4.3.1	INHERITS_FROM	3-19
3.2.4.3.2	COMPONENT	3-20
3.2.4.3.3	KEYWORDS	3-20
3.2.4.3.4	RELATION	3-21
3.2.4.4	REPRESENTATIONAL ATTRIBUTES.....	3-22
3.2.4.4.1	DATA_TYPE.....	3-22
3.2.4.4.2	ENUMERATION ATTRIBUTES	3-22
3.2.4.4.3	RANGE	3-24
3.2.4.4.4	TEXT_SIZE	3-24
3.2.4.4.5	CONSTANT_VALUE	3-24
3.2.5	USER-DEFINED ATTRIBUTES	3-25
3.2.5.1	MECHANISM FOR NEW ATTRIBUTE DEFINITIONS	3-25
3.2.5.2	REGISTRATION OF USER-DEFINED ATTRIBUTES	3-27
3.3	RELATIONSHIP RULES.....	3-27
3.3.1	REFERENCE TO AN EXTERNAL DEDSL DICTIONARY	3-27
3.3.2	COMPOSITION RELATION.....	3-27
3.3.3	INHERITANCE.....	3-28
3.3.4	OTHER KINDS OF RELATION	3-28
3.4	IMPLEMENTATION GUIDELINES.....	3-28
4	DEDSL CONFORMANCE: ABSTRACT DEDSL (ADID = CCSD0011)	4-1

5 ANNEX A: DEDSL EXAMPLESA**5.1 COMMUNITY DEDA.1****5.2 DICTIONARY ASSOCIATED WITH PRODUCT_X.....A.2****6 ANNEX B: INFORMATIVE REFERENCES..... B**Figures Page

FIGURE 2-1: ORGANISATION OF TWO RELATED DATA PRODUCTS 2-2

Tables Page

TABLE 2-1: COMPARISON OF COMMUNITY DED AND PRODUCT DED DESCRIPTIONS 2-4

TABLE 3-1: GENERAL DESCRIPTORS 3-2

TABLE 3-2: DEDSL DICTIONARY ATTRIBUTES..... 3-9

TABLE 3-3 : DEDSL DATA ENTITY ATTRIBUTES..... 3-10

Examples Page

EXAMPLE 3-1: ELEMENTARY TYPE..... 3-5

EXAMPLE 3-2: SET OF ELEMENTARY TYPES 3-5

EXAMPLE 3-3: USE OF ENTITY_TYPE 3-5

EXAMPLE 3-4: DEFINING USER-DEFINED ATTRIBUTE 3-26

EXAMPLE 3-5: USAGE OF A USER-DEFINED ATTRIBUTE..... 3-26

1 INTRODUCTION

1.1 PURPOSE AND SCOPE

The purpose of this Recommendation is to define a language for specifying a dictionary which describes the semantics of data entities - it does not define a specific dictionary.

A dictionary is understood as a mechanism that is able to organise a set of information in a consistent and easily understandable manner, and it is commonly used by humans to look up the meaning of words used in natural languages. Similarly, a data entity dictionary is used by humans and systems to look up the general meaning, and other attributes, of data entities used in the definition and generation of data products.

This Recommendation defines the concepts of name, definition, units, and a small set of other standard attributes so they may be used consistently in the formation of data entity dictionaries. The method used to define standard attributes can also be used to extend the set of attributes beyond the standard ones provided within this Recommendation. Given the wide variety of data entities that may need to be described, only a few of the attributes are made mandatory by this Recommendation.

Several classes of data entities are defined. These classes allow us to make a clear difference between the abstract data entities - the concepts - and the concrete data entities - the data fields in a data product -. This recommendation also provides the mechanisms allowing the definition of data fields in terms of abstract data entities.

This Recommendation is strongly based on the ISO/IEC Specification and standardization of data elements (see Reference [8]) with which it is widely consistent for all semantic aspects but not for the syntactic part. Independently of this consistency, this Recommendation is intended to be self-understandable.

1.2 APPLICABILITY

This Recommendation is intended to be used for example:

- By data producers to construct dictionaries that describe, in a more formal manner, data entities within their data products.
- By data users to understand data received from data producers who have used this Recommendation to construct their dictionaries.
- By an organisation that mandates the attributes used to define each entity description in dictionaries used within that organisation.
- By a particular community, such as Earth observation, space physics, archives, etc., to establish a degree of standardisation for the contents of any data dictionary associated or not with a data product. This would be done by using this Recommendation to define a community-wide data dictionary.
- By organisations and communities to exchange the contents of a data dictionary in a standardised manner, i.e., to facilitate interoperability.

1.3 RATIONALE

The same data entity can be physically represented within many different formats, including generic formats such as Hierarchical Data Format (HDF) (see Reference [E6]) and Common Data Format (CDF) (see Reference [E7]), or in native formats. However there is information about the data entities, such as their general meaning and other semantic attributes, which are independent of the physical representation.

Part of this information can be expressed in a data dictionary and this dictionary can be expressed in many different ways. For example, it may be expressed in natural language paragraphs within a document that accompanies a data product. It may be partially expressed by attributes defined within generic or native data product formats, while the rest may be in other documents. The concepts used in the description of the data entities may vary widely, or subtly. These concepts may not be documented at all.

Therefore, individuals and organisations that need to receive and understand a variety of data products may waste considerable effort in attempting to understand the data entities comprised by each data product. This also greatly hinders the use of generic tools that can assist in the recognition and presentation of this information in a way that various individuals and organisations find most understandable.

To begin to address these issues across broad organisation and community disciplines, it is first necessary to define a set of standard concepts that can be used in the formation of data dictionaries within those disciplines, and in the mapping of different dictionary concepts between disciplines. To facilitate the creation of generic tools, it is also necessary to define a standard representation for the standard concepts. This Recommendation defines a small number of concepts, in terms of attribute descriptors and standard attributes which are intended to be broadly applicable. It also defines an optional representation for defining and expressing these attributes.

Finally, this Recommendation provides additional standardised functionalities allowing the expression of relationships between entities in a dictionary and definitions of inter-relationships between dictionaries.

1.4 DOCUMENT STRUCTURE

This document defines the abstract definition of the semantic information that is required to be conveyed and presents the specification in a layered manner (attributes, entities, dictionaries). This is done so that the actual technique used to convey the information is independent of the information content and therefore the same abstract standard can be used within different formatting environments. This also permits the semantic information to be translated to different representations as may be needed when data are transferred across different domains.

In summary, the document is structured as follows:

- Section 2 provides an overview of the data entity dictionary concept and describes through examples how this Recommendation may be used. It also provides a context enabling a better understanding of the standard specified in Section 3.
- Section 3 specifies the abstract semantic description technique, including what information must be conveyed and when it is applicable.
- Section 4 discusses the level of conformance of the DEDSL Recommendation in relation to the CCSDS Control Authority registration of this Recommendation.
- Annex A provides examples implementing the concepts described in Section 3.
- Annex B provides a list of references that may be valuable to the user of this Recommendation as background material or as implementation guidelines for using this Recommendation.

1.5 DEFINITIONS

1.5.1 ACRONYMS AND ABBREVIATIONS

This sub-section defines the acronyms and abbreviations which are used throughout this Recommendation:

ADID	Authority and Description Identifier
ASCII	American Standard Code for Information Interchange
CCSDS	Consultative Committee for Space Data Systems
CDF	Common Data Format
DDL	Data Definition Language
DED	Data Entity Dictionary
DEDSL	Data Entity Dictionary Specification Language
EAST	Enhanced Ada SubseT
FITS	Flexible Image Transport System
HDF	Hierarchical Data Format
ID	IDentifier
ISO	International Organization for Standardization
LVO	Label Value Object
MACAO	Member Agency Control Authority Office
SFDU	Standard Formatted Data Unit

1.5.2 GLOSSARY OF TERMS

For the purpose of this document the following definitions apply:

Attribute	An attribute is a piece of information that describes another piece of data; this information characterises or enhances the understanding of the data being described. Attributes are the primary focus of this Recommendation in that they are used to define the semantics of data. This Recommendation specifies a standard method for defining standard and user-defined attributes so as to produce much less ambiguous definitions.
Attribute descriptor	Descriptors are the means by which Attributes are defined, registered and controlled.
Attribute value	A representation of an instance of an Attribute.
Data entity	A data entity is the piece of data which is to be described by using attributes and the values of the attributes. A data entity can be a data field in a data product or an abstract concept usable for the definition of other data entities. A data entity can be of an elementary type, such as integer or real; it can also be a composite type consisting of a structure of various other elementary and composite entities.
Data entity dictionary	A data entity dictionary is a collection of a number of semantic definitions of various data entities. Each data entity described in the dictionary is described in terms of attributes and attribute values. Data entity dictionaries may be just for a single product, i.e., all the data entities within a single product are described in a corresponding single dictionary, or the data entity dictionary may be a discipline-oriented dictionary that holds a number of previously defined data entity definitions which may be used by data designers and users as references. Some parts of a dictionary are optional. In practical terms the dictionary could be either a file or Standard Formatted Data Unit (SFDU) Label-Value Object (LVO) value field (see References [1] and [E4]).
Data product	A collection of one or more data items that are packaged for or by a specific application.
Data type	In the context of this document, data type is an indication of the form of the data entity, such as integer, real, text, enumerated, composite.
Descriptor Name	An Identifier being the name of the descriptor. All descriptor names are not case-sensitive.
Descriptor Type	The definition of the descriptor value: text, identifier, integer, ...
Entity Type	The type of a data entity.

Enumerated	A list containing a restricted number of discrete values, where each discrete value is named and unique within the list. Thus, it corresponds to a list of identifiers.
Identifier	<p>An unquoted sequence of ASCII characters (see Reference [2]) for which this Recommendation defines the following restrictions:</p> <ul style="list-style-type: none"> - no white spaces are allowed, - it begins with a letter and can be followed by letters, digits or the underline character, - it ends with a letter or a digit, - all characters must be in a visibly displayable form. <p>An Identifier must be unique within the scope of a DEDSL dictionary. Its size can be specified. Its case-sensitivity is defined by the global DEDSL attribute CASE_SENSITIVITY.</p>
Integer	The set of integer values. It can eventually be defined more precisely by specifying a range (minimum and maximum bounds).
Inter-operability Rules	Set of rules enabling an easier exchange of dictionaries using different DEDSL implementations.
Real	The set of real values. It can eventually be defined more precisely by specifying a range (minimum and maximum bounds).
Semantics	Semantic information that describes the meaning rather than the physical representation of data. Semantics potentially cover a very large domain, from the simple domain such as the units of one data entity to a more complex one, such as the relationship between a data entity and another one.
Standard attribute	A standard attribute is one of those defined within this Recommendation. When the DEDSL is used there are standard attributes that are mandatory and other ones which are optional. Attribute names defined in this Recommendation cannot be redefined if the semantic information is to be in conformance with this Recommendation.
Syntax	Syntax is the definition of the physical representation of data. It includes the structural arrangement of the fields within the data and the physical hardware representation. It results in a clear understanding of the abstract values of the data.

Text	<p>A delimited sequence of characters.</p> <p>The set of allowed characters is defined thanks to the DEDSL dictionary attribute: TEXT_FIELD_CHARACTER_SET.</p> <p>White space characters are allowed.</p> <p>Its size can be specified.</p>
User defined attribute	<p>A user-defined attribute is an attribute that is defined by a particular user or project and after definition is then used in the same manner as a 'standard attribute'.</p>
White space	<p>White space is defined as consisting of the equivalent of the ASCII characters line feed (0A_{hex}), carriage return (0D_{hex}), horizontal tab (09_{hex}), vertical tab (0B_{hex}), form feed (0C_{hex}) and space (20_{hex}) (see Reference [2]).</p>

1.5.3 NOMENCLATURE

The following conventions apply throughout this Recommendation:

- a) The words «shall» and «must» imply binding and verifiable specification;
- b) The word «should» implies an optional, but desirable, specification;
- c) The word «may» implies an optional specification;
- d) The words «is», «are» and «will» imply statements of fact.

1.6 REFERENCES

The following documents contain provisions (through references within this text) which constitute provisions of this Recommendation. At the time of the publication the indicated editions were valid. All documents are subject to revision, and users of this Recommendation are encouraged to investigate the possibility of applying the most recent editions of the documents indicated below. The CCSDS Secretariat maintains a register of currently available CCSDS Recommendations.

- [1] *Standard Formatted Data Units - Structure and Construction Rules.* Recommendation for Space Data System Standards, CCSDS 620.0-B-2. Blue Book. Issue 2. Washington, D.C.: CCSDS, May 1992.
- [2] *ASCII Encoded English (CCSD0002).* Recommendation for Space Data System Standards, CCSDS 643.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, November 1992.
- [3] *Information Processing - Representation of numerical values in character strings for information interchange.* ISO 6093-1985. Geneva: ISO, 1985.
- [4] *Standard Formatted Data Units -- Control Authority Procedures.* Recommendation for Space Data System Standards, CCSDS 630.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, June 1993.
- [5] *Code for the representation of names of languages.* ISO 639-1988. Geneva : ISO, 1988
- [6] *Information Processing - 8-Bit Single-Byte Coded Graphic Character Sets—Part 1: Latin Alphabet No. 1.* International Standard, ISO 8859-1:1987. Geneva: ISO, 1987.
- [7] *Information Processing - Representation of SI and other units in systems with limited character sets.* ISO 2955-1983. Geneva: ISO, 1983.
- [8] *Information Technology - Specification and standardisation of data elements - part 3: Basic attributes of data elements.* ISO/IEC 11179-3:1994(E). Geneva: ISO, 1994.

2 OVERVIEW

As discussed in the purpose and scope a data entity dictionary is used by humans and systems to look up the general meaning and other attributes of data entities used in the definition and generation of data products. This section discusses some of the primary uses of Data Entity Dictionaries (DED) and presents a basic example of DEDSL usage.

Other detailed examples are given in Annex A.

2.1 USES OF DATA ENTITY DICTIONARIES

2.1.1 PRODUCT DATA ENTITY DICTIONARIES

So that its contained data can be extracted, a data product can appear with a formatting standard (e.g., Flexible Image Transport System [FITS]), or a self-describing format (e.g., CDF, HDF, etc.) or a Data Definition Language (e.g., EAST).

This syntactic information may not be easy to understand and a formal definition of additional semantics may be necessary, which leads to the definition of a product Data Entity Dictionary.

This Recommendation focuses on developing standard names and descriptions for the concepts required for Data Entity Dictionaries and formally defines the concepts of name, definition, units, and a small set of other attributes so they may be used consistently in the formation of data entity dictionaries. A method is also provided permitting the set of attributes to be extended beyond the standard ones provided within this Recommendation.

This formal definition would enable the definition of generic tools to assist producers in creating documented products and to assist consumers in understanding the products they receive.

Considering for example the data product **PRODUCT_X** in Figure 2-1, the following semantic information seems important to be known. The entity **ACQ_STATION** is the identifier of the station which has acquired the data. The entity **ACQ_TIME** represents the date and time of the data acquisition. The entity **CENTRE_COORD** is meaningful as being made up of two component entities **LATITUDE** and **LONGITUDE**, considered as the centre coordinates for the associated data array **DATA_1**. The entity **DATA_1** represents an image taken from spacecraft W. Moreover additional information can be given to precise the kinds of the values of the **LATITUDE** and **LONGITUDE** used as the centre coordinates. The values of **LATITUDE** are expressed according to the Equator from -90.000 to +90.000 while the values of **LONGITUDE** are expressed according to Greenwich from -180.00 to +180.00.

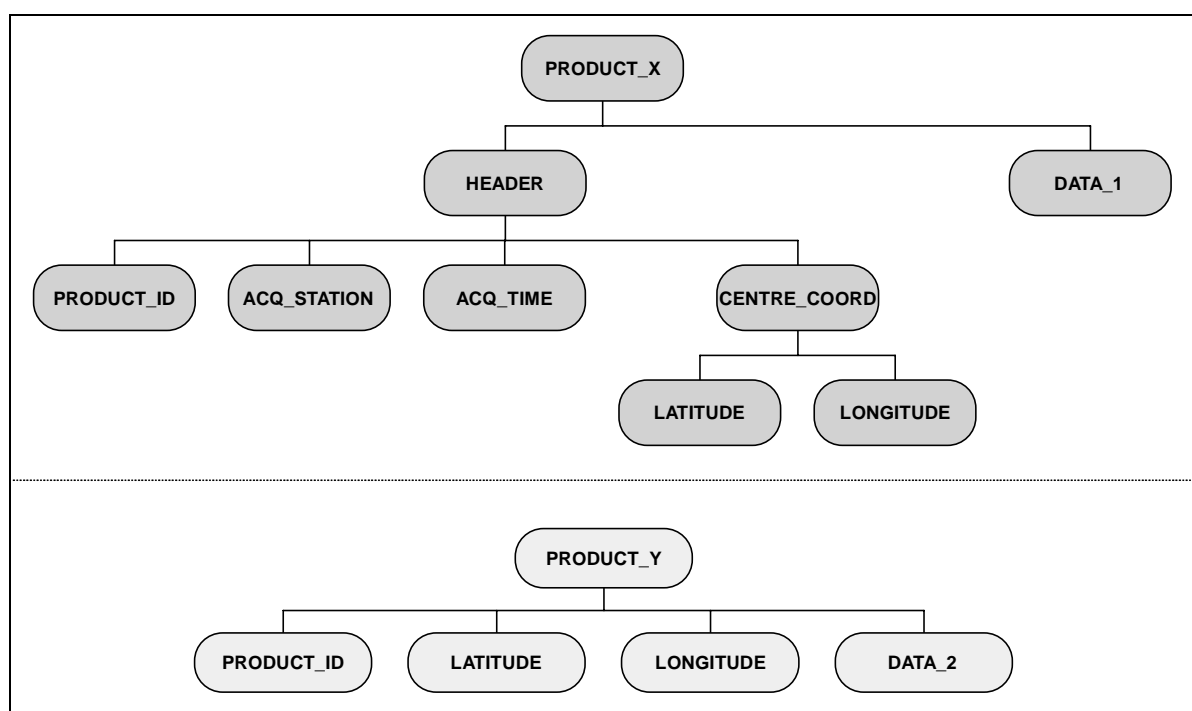


Figure 2-1: Organisation of Two Related Data Products

The techniques discussed in Section 3 of this Recommendation can be applied to any named data entity so a project or data designer can associate attributes with that entity. Using the example above, DEDSL attributes such as the name, the definition can be used for **LATITUDE** to express the semantics mentioned above in a product DED.

The information relative to **LATITUDE** may apply to other entities and then the data entity **LATITUDE** appears as a concept or set of semantic information which can be re-used for the semantic definition of these other entities.

2.1.2 COMMUNITY DATA ENTITY DICTIONARIES

A community DED is a means for an organisation to gain some degree of control/standardisation over the data descriptions created by member data producers. Unlike product dictionaries, these community DED are not used in conjunction with a specific product description technique. They are independent of the specific implementation of products. Examples of uses of community DED include:

- The creation of a standard data entity dictionary by an organisation that mandates the attributes defining each entity description in dictionaries used within that organisation.
- The creation of a community DED by a particular community (e.g., planetary science, astrophysics, etc.), to establish a degree of standardisation for the contents of any data dictionary associated with a data product from that community.

This recommendation provides a foundation for the creation of community DED by providing a basic set of concepts for data entity description and also provides the formal methods to describe relationships among data entities in a single DED or among multiple DED.

The data entities **LATITUDE** and **LONGITUDE** can be normalized to have the same semantic information associated with the data entities being latitudes and longitudes.

The example shown in the next section demonstrates how product and community DED can be built using this recommendation about the relationships among named entities or DED.

2.2 APPLICATION OF THE DEDSL

The two main application areas for the DEDSL are the following:

- To build up and define a community DED,
- To describe semantically a data product.

The project or the data designer may consider that a community dictionary is necessary since there are several data products related to the same kind of data. They may decide that the data dictionary should include the following entities which frequently appear: **PRODUCT_ID**, **ACQ_STATION**, **ACQ_TIME**, **LATITUDE** and **LONGITUDE**.

Then a description conforming to the DEDSL is given for each of the entities.

The purpose of a community dictionary is to provide, across different data products, a standard definition of data entities.

Table 2-1 shows an example of mapping of community DED entries into data product DED entries for the data products shown in Figure 2-1 according to the choices made by the project or data designers.

Entity Name	Data Type	Community DED	PRODUCT_X Description	PRODUCT_Y Description
HEADER	Composite	no	yes	no
PRODUCT_ID	Text	yes	yes	yes
ACQ_STATION	Enumeration	yes	yes	no
ACQ_TIME	Composite	yes	yes	no
CENTRE_COORD	Composite	no	yes	no
LATITUDE	Real	yes	yes	yes
LONGITUDE	Real	yes	yes	yes
DATA 1	Composite (Array of 16-bit integers)	no	yes	no
DATA 2	Composite (Array of real numbers)	no	no	yes

Table 2-1: Comparison of Community DED and Product DED Descriptions

When the project defines a data product (**PRODUCT_X** in Figure 2-1), the community dictionary is referenced in the product dictionary, and the **HEADER** entity is defined on the basis of **PRODUCT_ID**, **ACQ_STATION** and **ACQ_TIME**, which inherit from the properties of the corresponding data entities in the community dictionary. The **CENTRE_COORD** entity is defined on the basis of the **LATITUDE** and **LONGITUDE** entities which inherit from the properties of the corresponding data entities in the community dictionary. The **DATA1** entity is also defined to carry the image.

In the other product in Figure 2-1, **PRODUCT_Y**, the **PRODUCT_ID**, **LATITUDE** and **LONGITUDE** entities inherit from the community dictionary data entities. The **DATA2** entity is defined to carry the actual value.

2.3 REGISTERING DED

A community DED can be registered at different levels:

- In the framework of any organization dealing with data of a particular domain or project,
- Internally within an agency,
- Within the CCSDS community.

Whenever a member agency considers that one of its particular community DED corresponds to the needs of other agencies, it may submit its DED to the CCSDS Control Authority for registration according to the Control Authority Procedures mentioned in the corresponding Recommendation (References [4] and [E5]).

The CCSDS Control Authority provides operational services for registration, archiving and dissemination of data description data.

3 ABSTRACT STANDARD

The semantic information required for describing a data entity is seen as a collection of attributes. Each attribute describes a particular semantic characteristic of the data entity.

A data entity attribute shall be described in a standard way in order to achieve consistency among data entity dictionaries.

Section 3.1 gives the list of the descriptors of a data entity attribute.

Section 3.2 presents some concept definitions ('standard attributes', 'user-defined attributes', kinds of data entities, relationships among data entities, collection of data entities, ...). It also identifies the attributes that shall be used to describe the semantics of a data entity and those to be used to describe a data entity dictionary.

Section 3.3 refines the relationship rules.

Section 3.4 provides some implementation guidelines.

3.1 DESCRIPTORS OF A DATA ENTITY ATTRIBUTE

Data entity attributes shall be registered and controlled in a standard way in order to achieve consistency in the exchange of information on data entities among data entity dictionaries and to enable the comparison of data entities used in different management environments.

This recommendation defines a list of general descriptors for describing a data entity attribute. The descriptor names are not case-sensitive.

A descriptor is either mandatory, conditional or optional when a data entity attribute is defined. A descriptor can only appear once in a data entity attribute description, the attribute_value_example excepted.

The descriptors defined within the scope of this document are:

- attribute_name,
- attribute_definition,
- attribute_obligation,
- attribute_condition,
- attribute_maximum_occurrence,
- attribute_value_type,
- attribute_maximum_size,
- attribute_enumeration_values,
- attribute_comment,
- attribute_inheritance,
- attribute_default_value,
- attribute_value_example.

The following table gives the set of general descriptors that are defined by this recommendation. The obligation column indicates whether a descriptor is mandatory (M), conditional (C) or optional (O).

descriptor of attribute	obligation
attribute_name	M
attribute_definition	M
attribute_obligation	M
attribute_condition	C
attribute_maximum_occurrence	M
attribute_value_type	M
attribute_maximum_size	C
attribute_enumeration_values	C
attribute_comment	O
attribute_inheritance	M
attribute_default_value	O
attribute_value_example	O

Table 3-1: General descriptors

3.1.1 ATTRIBUTE_NAME

Purpose Label assigned to a data entity attribute.

Descriptor Name The standard term to be used for this descriptor is:
ATTRIBUTE_NAME

Obligation This descriptor is mandatory.

Descriptor Type The value of this descriptor is an Identifier which is not case-sensitive.

Interoperability Rule The maximum length proposed for the value of this descriptor is 40 characters.

3.1.2 ATTRIBUTE_DEFINITION

- Purpose** The definition is required to give the description of the data entity attribute. This definition is intended for human readership and therefore any information that increases the understanding of the identified attribute should be included.
- Descriptor Name** The standard term to be used for this descriptor is:
ATTRIBUTE_DEFINITION
- Obligation** This descriptor is mandatory.
- Descriptor Type** The definition is represented as a Text.
- Interoperability Rule** It is intended that the value of this descriptor can be of a significant length and hence provide a description of the attribute as complete as possible.
The maximum length proposed for the value of this descriptor is 8000 characters.

3.1.3 ATTRIBUTE_OBLIGATION

- Purpose** Descriptor indicating whether a data entity attribute shall always be present or only sometimes according to specified conditions.
- Descriptor Name** The standard term to be used for this descriptor is:
ATTRIBUTE_OBLIGATION
- Obligation** This descriptor is mandatory.
- Descriptor Type** This descriptor is an Enumerated with three discrete values corresponding to the following cases:
- mandatory: the data entity attribute shall always be present.
 - conditional: the data entity attribute shall be present if conditions specified by the descriptor ATTRIBUTE_CONDITION occur for the same data entity attribute.
 - optional: the data entity attribute may be present or not.
- Interoperability Rule** The coding values are M or mandatory, C or conditional and O or optional.

3.1.4 ATTRIBUTE_CONDITION

- Purpose** Descriptor indicating the circumstances under which a data entity attribute shall be present.
- Descriptor Name** The standard term to be used for this descriptor is:
ATTRIBUTE_CONDITION
- Obligation** This descriptor is conditional.
It shall be present if the ATTRIBUTE_OBLIGATION descriptor of the same data entity attribute has the value 'conditional'.
- Descriptor Type** The value is represented as a Text.
- Interoperability** The maximum length proposed for the value of this descriptor is 8000 characters.
- Rule** It does not exclude the possibility of defining a particular formalism for the text itself, enabling the automatic processing of the text.

3.1.5 ATTRIBUTE_MAXIMUM_OCCURRENCE

- Purpose** Descriptor specifying the maximum number of instances which the data entity attribute may have in the specification of a data entity.
- Descriptor Name** The standard term to be used for this descriptor is:
ATTRIBUTE_MAXIMUM_OCCURRENCE.
- Obligation** This descriptor is mandatory.
When omitted, it assumes that its default value is 1.
- Descriptor Type** The value of this descriptor is an integer or the character 'n' specifying that there is no upper limit on the number of times that the data entity attribute may occur.
This recommendation defines a default value of 1 for this descriptor.

3.1.6 ATTRIBUTE_VALUE_TYPE

- Purpose** Descriptor specifying a set of distinct values for representing the attribute value.
- Descriptor Name** The standard term to be used for this descriptor is:
ATTRIBUTE_VALUE_TYPE.
- Obligation** This descriptor is mandatory.
- Descriptor Type** The descriptor value can be of elementary type and then must have one of the following values: Enumerated, Integer, Real, Text, Identifier.
The descriptor value can be of the type of the value of the data entity and then must be the word: Entity_Type.
The descriptor value can also be made up of an ordered list of elementary types or be specified as a choice among a list of elementary types.

The following example presents an attribute of elementary type:

For the definition of the attribute called NAME that identifies the data entity, the ATTRIBUTE_VALUE_TYPE specifies that a value of this attribute is an identifier.
ATTRIBUTE_VALUE_TYPE : Identifier

Example 3-1: Elementary type

The following example presents an attribute type defined as a set of two elementary types:

For the definition of the attribute called ALIAS that defines an alternative name of a data entity, the ATTRIBUTE_VALUE_TYPE specifies that a value of this attribute is composed by the alternative name (of the type **Identifier**) and the context in which this alias is used (of the type **Text**).
ATTRIBUTE_VALUE_TYPE : (Identifier, Text)

Example 3-2: Set of elementary types

The following example illustrates the use of Entity_Type:

For the definition of the attribute called SPECIFIC_INSTANCE that provides a specific value of the data entity occurrence, the ATTRIBUTE_VALUE_TYPE specifies that a value of this attribute is composed by a specific value (of the type **Entity_Type**) and the meaning and context of this specific value (of the type **Text**).
ATTRIBUTE_VALUE_TYPE : (Entity_Type, Text)

Example 3-3: Use of Entity_Type

3.1.7 ATTRIBUTE_MAXIMUM_SIZE

Purpose Descriptor specifying the maximum number of characters for representing the value of the attribute.

Descriptor Name The standard term to be used for this descriptor is:
 ATTRIBUTE_MAXIMUM_SIZE

Obligation This descriptor is conditional.
 It shall be present if the ATTRIBUTE_VALUE_TYPE is Identifier or Text.
 It may be present for composite attributes made up of Identifiers or Texts. It then represents the maximum size allowed for the entire set of components.

Descriptor Type The value of this descriptor is an integer value.

3.1.8 ATTRIBUTE_ENUMERATION_VALUES

Purpose	Descriptor specifying the distinct and discrete values of the attribute.
Descriptor Name	The standard term to be used for this descriptor is: ATTRIBUTE_ENUMERATION_VALUES.
Obligation	This descriptor is conditional. It shall be present if the ATTRIBUTE_VALUE_TYPE is Enumerated.
Descriptor Type	The value of this descriptor is a list of Identifiers.
Interoperability Rule	The maximum length proposed for each Identifier is 40 characters.

3.1.9 ATTRIBUTE_COMMENT

Purpose	Descriptor providing information which is not directly required to understand the meaning of the attribute, but which could still assist the user of the attribute in some manner. It may also contain examples.
Descriptor Name	The standard term to be used for this descriptor is: ATTRIBUTE_COMMENT
Obligation	This descriptor is optional.
Descriptor Type	The definition is represented as a Text.
Interoperability Rule	The maximum length proposed for the value of this descriptor is 8000 characters.

3.1.10 ATTRIBUTE_INHERITANCE

Purpose	Descriptor providing information about the inheritance rules for the attribute.
Descriptor Name	The standard term to be used for this descriptor is: ATTRIBUTE_INHERITANCE
Obligation	This descriptor is mandatory.
Descriptor Type	This descriptor is an Enumerated with three discrete values corresponding to the following cases: <ul style="list-style-type: none"> • NOT_INHERITED: it applies to two cases : either the notion of inheritance is not meaningful or the attribute cannot be inherited. It then can be defined locally at the level of the data entity description which inherits from another data entity description. • INHERITED: the attribute is inherited and cannot be defined locally. • INHERITED_AND_ENRICHABLE: the attribute is inherited and can be further defined locally at the level of the data entity description which inherits the attribute from another data entity description.

3.1.11 ATTRIBUTE_DEFAULT_VALUE

- Purpose** Descriptor providing a default value for the attribute.
- Descriptor Name** The standard term to be used for this descriptor is:
ATTRIBUTE_DEFAULT_VALUE.
- Obligation** This descriptor is optional.
- Descriptor Type** The format of this descriptor must conform to the type of the attribute that it illustrates, i.e. must be compliant with the value of the descriptor
ATTRIBUTE_VALUE_TYPE.

3.1.12 ATTRIBUTE_VALUE_EXAMPLE

- Purpose** Descriptor providing an example of a value of the attribute as it would appear in a data entity definition.
- Descriptor Name** The standard term to be used for this descriptor is :
ATTRIBUTE_VALUE_EXAMPLE
- Obligation** This descriptor is optional and may appear any number of times demonstrating a different example in each case.
- Descriptor Type** The format of this descriptor must conform to the type of the attribute that it illustrates, i.e. must be compliant with the value of the descriptor
ATTRIBUTE_VALUE_TYPE.

3.2 DATA ENTITY ATTRIBUTES AND DICTIONARY ATTRIBUTES

3.2.1 CONCEPT DEFINITIONS

A specification or semantic description of a data entity consists of a set of attributes.

Some of these attributes which are considered as frequently needed are defined in this recommendation as **standard attributes** (see sections 3.2.2 to 3.2.4). Additional attributes called **user-defined attributes** may be required according to the user's needs. The standard attributes are of character type. Note that this does not forbid user-defined attributes with non-character values.

The method to define them is described in section 3.2.5.

A **dictionary** is a collection of data entity descriptions. In this view, a dictionary can be seen as a '**dictionary entity**' that will also be described with attributes (called **dictionary attributes**).

Standard attributes will be used to describe:

- some features of the dictionary as a whole (see section 3.2.3),
- the data entities in the dictionary (see section 3.2.4).

Three classes of data entities are defined:

- **concept class**: for a data entity described independently from any instance in a data product (e.g. in a 'domain' dictionary), and corresponding to a re-usable data entity model from which other data entities may inherit the attributes,
- **data field class**: for a data entity in a data product, having its own specific attributes,
- **constant class**: for a named constant value used within a dictionary but not being part of the data themselves. Such a class enables to fix the values which will be used by several projects or within a domain (astronomy constants, image size, ...). It corresponds to metadata-level information.

It is important to make clear in a dictionary what is the class of the defined entities as these entities will be used differently according to their class, although they will be described with the same attributes.

Consequently, each data entity description shall include a specific attribute named '**CLASS**' intended to indicate the class of the entity being described.

The standard attributes are described using the descriptors previously defined and can be:

- **mandatory**: always required,
- **conditional**: required under specified conditions,
- **optional**: allowed but not required.

These attributes are classified into five categories:

- **identifying attributes** that are applicable for the identification of a data entity,
- **definitional attributes** that describe core semantic aspects of a data entity,
- **relational attributes** that describe associations among data entities,
- **representational attributes** that describe representational aspects of a data entity,
- **administrative attributes** that describe management and control aspects of a data entity.

Relational attributes describe the relationships which may appear among data entities and the following kinds of relationships have been distinguished:

- the **composition** relationship: a data entity may be composite, that is, it can be made up of other data entities considered as components,
- the **inheritance** relationship: a data entity may be quite identical to another one; therefore, a concept data entity may be defined and the two data entities then refer to this concept from which they inherit the different attributes. However, they may have their own additional attributes or further define the inherited attributes.
- the **reference** to another data entity dictionary, where needed concept data entities have already been defined.
- **other relationships**: other possible relations.

3.2.2 GENERAL VIEW OF THE STANDARD ATTRIBUTES

3.2.2.1 STANDARD ATTRIBUTES FOR DEDSL DICTIONARIES

The following table gives for each category the standard attributes defined by this recommendation for the DEDSL Dictionary. The obligation column indicates whether an attribute is mandatory (M), conditional (C) or optional (O).

Attribute Category	Name of data entity attribute	Obligation
Identifying	DICTIONARY_NAME	M
Definitional	DICTIONARY_DEFINITION	O
Relational	EXTERNAL_DICTIONARY_REFERENCE	O
Representational	TEXT_FIELD_CHARACTER_SET	M
	CASE_SENSITIVITY	M
	DICTIONARY_LANGUAGE	O
Administrative	DICTIONARY_VERSION	O
	DICTIONARY_ADID	O
	DEDSL_VERSION	M

Table 3-2: DEDSL dictionary attributes

3.2.2.2 STANDARD ATTRIBUTES FOR DATA ENTITIES

The following table gives for each category the standard attributes that are defined by this recommendation for data entities. The obligation column indicates whether an attribute is mandatory (M), conditional (C) or optional (O).

Attribute Category	Name of data entity attribute	Obligation
Identifying	NAME	M
	ALIAS	O
	CLASS	M
Definitional	DEFINITION	M
	SHORT_DEFINITION	O
	COMMENT	O
	UNITS	C
	SPECIFIC_INSTANCE	O
Relational	INHERITS_FROM	O
	COMPONENT	O
	KEYWORDS	O
	RELATION	O
Representational	DATA_TYPE	C
	ENUMERATION_VALUES	C
	ENUMERATION_MEANING	C
	ENUMERATION_CODING	C
	RANGE	O
	TEXT_SIZE	C
	CONSTANT_VALUE	O

Table 3-3 : DEDSL data entity attributes

3.2.3 DEDSL DICTIONARY ATTRIBUTES

3.2.3.1 DICTIONARY_NAME

Attribute_Name : **DICTIONARY_NAME**
 Attribute_Definition : Human readable name for the Data Entity Dictionary.
 Attribute_Obligation : Mandatory.
 Attribute_Value_Type : Identifier.
 Attribute_Maximum_Occurrence : 1.
 Attribute_Inheritance : NOT_INHERITED.

Interoperability Rule
 Attribute_Maximum_Size : 400.

3.2.3.2 DICTIONARY_DEFINITION

Attribute_Name : **DICTIONARY_DEFINITION**
 Attribute_Definition : Human readable definition for the Data Entity Dictionary.
 Attribute_Obligation : Optional.
 Attribute_Value_Type : Text.
 Attribute_Maximum_Occurrence : 1.
 Attribute_Inheritance : NOT_INHERITED.

Interoperability Rules
 Attribute_Maximum_Size : 400.
 Attribute_Comment : The value of this attribute is a free format text which can span a number of lines. It may include any number of white space characters.

3.2.3.3 EXTERNAL_DICTIONARY_REFERENCE

Attribute_Name	: EXTERNAL_DICTIONARY_REFERENCE
Attribute_Definition	: This attribute gives a reference to another Data Entity Dictionary whose concepts are reused in the current one. This reference is made up of the following information: <ul style="list-style-type: none"> - the name of the referenced external Data Entity Dictionary, - the optional dictionary identifier within a registration authority: it may also indicate an official reference registration (ADID) by a CCSDS Control Authority - the optional identifier of the registration authority.
Attribute_Obligation	: Conditional.
Attribute_Value_Type	: - (Identifier [, Identifier, Identifier])
Attribute_Maximum_Occurrence	: 'n'
Attribute_Condition	: This attribute is mandatory if a reference to a Data Entity Dictionary is made in the current data entity dictionary (in one of the inherits_from attributes).
Attribute_Value_Example	: (Dictionary_Name, CCSD0011, CCSDS)
Attribute_Inheritance	: NOT_INHERITED.
Interoperability Rule	
Attribute_Comment	The ADID must conform to the Recommendation (Reference [4]) and be an 8 character string.

3.2.3.4 TEXT_FIELD_CHARACTER_SET

Attribute_Name	: TEXT_FIELD_CHARACTER_SET
Attribute_Definition	: Character Set that is valid for the textual part of the attribute values within the dictionary.
Attribute_Obligation	: Mandatory.
Attribute_Value_Type	: Identifier.
Attribute_Maximum_Occurrence	: 1.
Attribute_Inheritance	: NOT_INHERITED.
Interoperability Rules	
Attribute_Maximum_Size	: 40
Attribute_Comment	: If the character set is not provided for the Data Entity Dictionary, then the default character set is given in the implementation.

3.2.3.5 CASE_SENSITIVITY

Attribute_Name	: CASE_SENSITIVITY
Attribute_Definition	: The value of this attribute enables to set the case sensitivity for all the Identifiers used as values given to the attributes within the dictionary.

Attribute_Obligation	: Mandatory. When omitted, its default value is NOT_CASE_SENSITIVE.
Attribute_Value_Type	: Enumerated
Attribute_Enumeration_Values	: (CASE_SENSITIVE, NOT_CASE_SENSITIVE)
Attribute_Maximum_Occurrence	: 1.
Attribute_Inheritance	: NOT_INHERITED.

3.2.3.6 DICTIONARY_LANGUAGE

Attribute_Name	: DICTIONARY_LANGUAGE
Attribute_Definition	: Natural language that is valid for the whole DEDSL dictionary. By default, the natural language is English.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: (Identifier, Text) where: <ul style="list-style-type: none"> - Identifier is provided in English and corresponds to the English name specified in ISO 639 (see Reference [5]). - Text is expressed in the specified language and corresponds to the original name as specified in ISO 639.
Attribute_Maximum_Occurrence	: 1.
Attribute_Value_Example	: (French, "Français").
Attribute_Inheritance	: NOT_INHERITED.
Interoperability Rule	
Attribute_Comment	The Identifier and the Text are both made up of 40 characters.

3.2.3.7 DICTIONARY_VERSION

Attribute_Name	: DICTIONARY_VERSION
Attribute_Definition	: Version of the Data Entity Dictionary.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: Text
Attribute_Maximum_Occurrence	: 1.
Attribute_Comment	: It corresponds to the issue and the revision of the current dictionary separated by a period.
Attribute_Value_Example	: "1.1"
Attribute_Inheritance	: NOT_INHERITED.

3.2.3.8 DICTIONARY_ADID

Attribute_Name	: DICTIONARY_ADID
Attribute_Definition	: the CCSDS Authority and Description Identifier (ADID) under which the Data Entity Dictionary has been registered at a Control Authority (see References [4] and [E5]).
Attribute_Obligation	: Optional.
Attribute_Value_Type	: Identifier.
Attribute_Maximum_Occurrence	: 1.
Attribute_Comment	The format of this parameter value is an unquoted ASCII string of eight consecutive Restricted ASCII characters that constitute a registered MACAO ADID (see References [4] and [E5]).
Attribute_Maximum_Size	: 8
Attribute_Inheritance	: NOT_INHERITED.

3.2.3.9 DEDSL_VERSION

Attribute_Name	: DEDSL_VERSION
Attribute_Definition	: Version of the DEDSL Recommendation that the Data Entity Dictionary conforms to. The version is indicated on the front cover of this Recommendation.
Attribute_Obligation	: Mandatory.
Attribute_Value_Type	: Text
Attribute_Maximum_Occurrence	: 1.
Attribute_Comment	: It corresponds to the issue and the revision separated by a period.
Attribute_Value_Example	: "2.2"
Attribute_Inheritance	: NOT_INHERITED.

3.2.4 DEDSL STANDARD ATTRIBUTES FOR DATA ENTITIES

3.2.4.1 IDENTIFYING ATTRIBUTES

3.2.4.1.1 NAME

Attribute_Name	: NAME
Attribute_Definition	: The value of this attribute may be used to link a collection of attributes with an equivalent identifier in, or associated with, the data entity. The value of this attribute may also be used by the software developer to name corresponding variables in software code or designate a field to be searched for locating particular data entities. The name shall be unique within a Data Entity Dictionary.
Attribute_Obligation	: Mandatory.
Attribute_Value_Type	: Identifier
Attribute_Maximum_Occurrence	: 1.
Attribute_Value_Example	: ACQ_STATION
Attribute_Inheritance	: NOT_INHERITED
Interoperability Rule	
Attribute_Maximum_Size	: 400.

3.2.4.1.2 ALIAS

Attribute_Name	: ALIAS
Attribute_Definition	: Single- or multi-word designation that differs from the given name, but represents the same data entity concept, followed by the context in which this name is applied. The value of this attribute provides an alternative identifier of the data entity that may be required for the purpose of compatibility with historical data or data deriving from different sources. For example, different sources may produce data including the same entities, but giving them different names. Through the use of this attribute it will be possible to define the semantic information only once. Along with the alternative identifier, this attribute value shall provide a description of the context of when the alternative identifier is used. The value of the alternative identifier can also be searched when an identifier used in a corresponding syntax description is not found within the NAME values. The value shall be unique within a Data Entity Dictionary.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: (Identifier, Text)
Attribute_Maximum_Occurrence	: 'n'.
Attribute_Value_Example	: (TIME_LINE, "used within the ground segment").
Attribute_Inheritance	: NOT_INHERITED.

Interoperability Rule

Attribute_Comment

The maximum length proposed for the context information text is 400 characters.

The maximum length proposed for the Identifier is the one defined for the **NAME**.

3.2.4.1.3 CLASS

Attribute_Name

: **CLASS**

Attribute_Definition

: The value of this attribute makes a clear statement of what kind of entity is defined by the current entity definition. This definition can be a concept definition, a data field definition, a constant definition.

Attribute_Obligation

: Mandatory.

Attribute_Value_Type

: Enumerated.

Attribute_Enumeration_Values

: (CONCEPT, DATA_FIELD, CONSTANT).

Attribute_Maximum_Occurrence

: 1.

Attribute_Inheritance

: NOT_INHERITED

3.2.4.2 DEFINITIONAL ATTRIBUTES

3.2.4.2.1 DEFINITION

Attribute_Name	: DEFINITION
Attribute_Definition	: Statement that expresses the essential nature of a data entity and permits its differentiation from all the other data entities. This attribute is intended for human readership and therefore any information that will increase the understanding of the identified data entity should be included. It is intended that the value of this attribute can be of significant length and hence provide a description of the data entity as complete as possible. The value of this attribute can be used as a field to be searched for locating particular data entities.
Attribute_Obligation	: Mandatory.
Attribute_Value_Type	: Text.
Attribute_Maximum_Occurrence	: 1.
Attribute_Comment	: The value of this attribute may include the same semantic information in natural language as the one carried in a more formal manner by other attributes. This is neither a requirement nor illegal, but the user must make sure that inconsistencies do not arise.
Attribute_Inheritance	: INHERITED_AND_ENRICHABLE.
Interoperability Rule	
Attribute_Maximum_Size	: 8000.

3.2.4.2.2 SHORT_DEFINITION

Attribute_Name	: SHORT_DEFINITION
Attribute_Definition	: Statement that expresses the essential nature of a data entity in a shorter and more concise manner than the statement of the mandatory attribute: DEFINITION. This attribute provides a summary of the more detailed information provided by the DEFINITION attribute. The value of this attribute can be used as a field to be searched for locating particular data entities. It is also intended to be used for display purposes by automated software, where the complete DEFINITION value would be too long to be presented in a convenient manner to users.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: Text.
Attribute_Maximum_Occurrence	: 1.
Attribute_Inheritance	: INHERITED_AND_ENRICHABLE.
Interoperability Rule	
Attribute_Maximum_Size	: 80.

3.2.4.2.3 COMMENT

Attribute_Name	: COMMENT
Attribute_Definition	: Associated information about a data entity. It enables to add information which does not correspond to definition information.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: Text.
Attribute_Maximum_Occurrence	: 'n'.
Attribute_Inheritance	: INHERITED_AND_ENRICHABLE.
Interoperability Rule	
Attribute_Maximum_Size	: 8000.

3.2.4.2.4 UNITS

Attribute_Name	: UNITS
Attribute_Definition	: Attribute that specifies the scientific units that should be associated with the value of the data entity so as to make the value meaningful in the 'real-world'.
Attribute_Obligation	: Conditional.
Attribute_Condition	: If the data entity that is being defined is of a scientific scalar type (Integer or Real) then this attribute is mandatory and may appear only once. If the data entity is non-scalar then the attribute shall not be specified. If the scalar type has no unit, e.g. a ratio, then the value of this attribute has to be NO_UNIT.
Attribute_Value_Type	: Text. The contents of the text must conform to the representation specified in ISO 2955 (see Reference [E10]). As detailed in ISO 2955, the following conventions apply when combining units: <ul style="list-style-type: none"> • Multiplication shall be indicated by a period (.), e.g., Pa.s to designate Pascal second, the unit of dynamic viscosity. • Negative exponents shall be indicated by following the unit directly with the numeric power preceded by a minus sign, e.g., m-3 to designate m⁻³ • Division shall be indicated by a solidus (/), e.g., m/s, or by expressing the denominator with a negative exponent, e.g., m.s-1 • Positive exponents shall be indicated by following the unit directly with the numeric power with no sign, e.g., m2 to designate m² Decimal multiples of units shall be indicated by the combination of a prefix representation (see Reference [E9]) immediately before the unit, e.g., kN to represent kilo Newtons.
Attribute_Maximum_Occurrence	: 1.
Attribute_Inheritance	: INHERITED.
Interoperability Rule	
Attribute_Maximum_Size	: 80.

3.2.4.2.5 SPECIFIC_INSTANCE

Attribute_Name	: SPECIFIC_INSTANCE
Attribute_Definition	: Attribute that provides a real-world meaning for a specific instance (a value) of the data entity being described. The reason for providing this information is so that the user can see that there is some specific meaning associated with a particular value instance that indicates something more than just the abstract value. For example, the fact that zero degree latitude is the equator could be defined. This means that the value of this attribute must provide both an instance of the entity value and a definition of its specific meaning.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: (Entity_Type, Text) There shall be two values associated with this attribute: an instance value (a literal or a constant name) and a specific meaning definition.
Attribute_Maximum_Occurrence	: 'n'
Attribute_Comment	: The values of the attribute can be used to enhance user interfaces and therefore user understanding. For example, instead of displaying to the user the abstract value of an entity, the 'system' could first check the DEDSL definition to see if there is a specific meaning for this abstract value, and if there is, display the specific meaning string instead. Likewise, a user could enter a meaning definition by name, e.g. equator , and the 'system' could automatically (via the DEDSL definition) translate this name to a specific instance value.
Attribute_Value_Example	: For a specific value of the Entity_Type DEGREE (being of the Real type), a specific instance could be: (0.0, "Greenwich").
Attribute_Inheritance	: INHERITED_AND_ENRICHABLE.
Interoperability Rule	
Attribute_Maximum_Size	: The maximum length proposed for the specific meaning text is 80 characters.

3.2.4.3 RELATIONAL ATTRIBUTES

3.2.4.3.1 INHERITS_FROM

Attribute_Name	: INHERITS_FROM
Attribute_Definition	: Gives the name of a concept from which the current entity description inherits attributes. This name must be the value of the NAME attribute found in an entity description whose CLASS attribute is set to CONCEPT. Referencing this data entity description means that all the values of its attributes having their ATTRIBUTE_INHERITANCE set to INHERITED or IINHERITED_AND_ENRICHABLE apply to the current description.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: (Identifier [, Identifier])

The first identifier refers to the name of a concept.

The second one gives the name of the external Data Entity Dictionary where the concept is defined.

The absence of the second identifier assumes that this referred concept is local to the current Data Entity Dictionary or, if not, that it is to be found in a default dictionary referenced through the EXTERNAL_DICTIONARY_REFERENCE attribute which must be unique in that case.

Attribute_Maximum_Occurrence	: 1.
Attribute_Comment	: This attribute is intended to enable reuse of concepts. Each data entity description belonging to the same concept should be qualified using the same value of this attribute.
Attribute_Value_Example	: (CCSDS_calendar_time, CCSDS_TIME_CODES)
Attribute_Inheritance	: NOT_INHERITED.
Interoperability Rule	
Attribute_Maximum_Size	: The maximum length proposed for the first identifier is 400 characters. The maximum length proposed for the second identifier is 40 characters.

3.2.4.3.2 COMPONENT

Attribute_Name	: COMPONENT
Attribute_Definition	: Name of a component, followed by the number of times it occurs in the composite data entity. The number of times is specified by a range.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: Identifier(a .. b). Where a is the minimum number of times the component occurs and b is the maximum number of times it occurs. a and b are integer literals or constant names. The following convention applies: the character 'n' indicates that there is no upper limit.
Attribute_Maximum_Occurrence	: 'n'.
Attribute_Comment	: This attribute can be used only for composite data entities (arrays, records or lists). When a composite data entity has no component defined, it means that there are not known yet, and will be specified later.
Attribute_Inheritance	: INHERITED_AND_ENRICHABLE.
Interoperability Rule	
Attribute_Maximum_Size	: The maximum length proposed for the Identifier is 400 characters.

3.2.4.3.3 KEYWORDS

Attribute_Name	: KEYWORDS
Attribute_Definition	: One or several significant words used for retrieving data entities.

Attribute_Obligation	: Optional.
Attribute_Value_Type	: Text.
Attribute_Maximum_Occurrence	: 1.
Attribute_Inheritance	: INHERITED_AND_ENRICHABLE.
Interoperability Rules	
Attribute_Comment	: This attribute can be used for recording keywords (search keys) associated with the data entity in question. Indeed, it enables the possibility of defining a particular formalism for the text and then an automatic processing of the text according to the needs.
Attribute_Maximum_Size	: 80

3.2.4.3.4 RELATION

Attribute_Name	: RELATION
Attribute_Definition	: This attribute is to be used to express a relationship between two entity definitions when this relation cannot be expressed using a precise standard relational attribute. In that case the relationship is user-defined and expressed using free text.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: (Text, Identifier [, Identifier])
Attribute_Maximum_Occurrence	: 'n'.
Attribute_Comment	: - The first attribute value provides the reader with the kind of relation that links the two related entities. - The second one is the name of the entity in relation with the one being defined. - The last one is used when the previous entity is described in an external Data Entity Dictionary to give the name of this dictionary.
Attribute_Value_Example	: ("Date of the current measurement", CCSDS_calendar_time, CCSDS_TIME_CODES)
Attribute_Inheritance	: INHERITED_AND_ENRICHABLE.
Interoperability Rule	
Attribute_Maximum_Size	: The maximum length proposed for the first attribute value is 80 characters. The maximum length proposed for the second one is 400 characters. The maximum length proposed for the last one is 40 characters.

3.2.4.4 REPRESENTATIONAL ATTRIBUTES

3.2.4.4.1 DATA_TYPE

Attribute_Name	: DATA_TYPE
Attribute_Definition	: It specifies the type of the data entity values. This attribute shall have one of the following values: Enumerated, Text, Real, Integer, Composite .
Attribute_Obligation	: Conditional.
Attribute_Condition	: This attribute must be present for a product data field definition and for a constant definition (CLASS attribute set to DATA_FIELD or CONSTANT) and is optional for a concept definition (CLASS attribute set to CONCEPT).
Attribute_Value_Type	: Enumerated.
Attribute_Enumeration_Values	: (Enumerated, Text, Real, Integer, Composite).
Attribute_Maximum_Occurrence	: 1.
Attribute_Comment	: This attribute defines the conceptual data type of the entity, it is not intended for specifying the physical representation of the entity. For example, an entity may be defined as an Integer; physically it may be encoded as a 16-bit 2's complement binary number or as an ASCII encoded decimal, but in both cases the DATA_TYPE would be Integer.
Attribute_Inheritance	: INHERITED.

3.2.4.4.2 ENUMERATION ATTRIBUTES

When a data entity belongs to an enumerated type (attribute DATA_TYPE set to Enumerated), it means that each possible value of the enumeration has:

- an enumerated value (an identifier as meaningful as possible),
- a meaning (which can be expressed in free text),
- a coding value (bit representation).

3.2.4.4.2.1 ENUMERATION_VALUES

Attribute_Name	: ENUMERATION_VALUES
Attribute_Definition	: The set of permitted values of the enumerated data entity.
Attribute_Obligation	: Conditional.
Attribute_Condition	: This attribute is mandatory if the DATA_TYPE is Enumerated. It is not applicable in any other case.
Attribute_Value_Type	: (Identifier, Identifier, ...).
Attribute_Value_Example	: (NOMINAL, CALIBRATION, OFF)
Attribute_Maximum_Occurrence	: 1.
Attribute_Inheritance	: INHERITED.

3.2.4.4.2.2 ENUMERATION_MEANING

Attribute_Name	: ENUMERATION_MEANING
Attribute_Definition	: Enables to give a meaning to each value given by the attribute ENUMERATION_VALUES.
Attribute_Obligation	: Optional.
Attribute_Condition	: This attribute is permitted if the DATA_TYPE is Enumerated. It is not applicable in any other case.
Attribute_Value_Type	: Text
Attribute_Value_Example	: (NOMINAL : equipment on and working, CALIBRATION : equipment under calibration tests, OFF : equipment off)
Attribute_Maximum_Occurrence	: 1.
Attribute_Inheritance	: INHERITED.

3.2.4.4.2.3 ENUMERATION_CODING

Attribute_Name	: ENUMERATION_CODING
Attribute_Definition	: Enables to give a coding to each value given by the attribute ENUMERATION_VALUES.
Attribute_Obligation	: Optional.
Attribute_Condition	: This attribute is permitted if the DATA_TYPE is Enumerated. It is not applicable in any other case.
Attribute_Value_Type	: Text
Attribute_Value_Example	: (NOMINAL : 0, CALIBRATION : 1, OFF : 2)
Attribute_Maximum_Occurrence	: 1.
Attribute_Inheritance	: INHERITED.

3.2.4.4.3 RANGE

Attribute_Name	: RANGE
Attribute_Definition	: The minimum bound and the maximum bound of an Integer or Real or Enumerated data entity.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: (Entity_Type, Entity_Type). The first specified value is the minimum bound (literal value or constant name) while the second one is the maximum bound (literal value or constant name).
Attribute_Maximum_Occurrence	: 1.
Attribute_Comment	: This attribute only applies for Integer, Real and Enumerated data entities.
Attribute_Value_Example	: For a Real data entity called DEGREE, the range could be as follows: (0.0, 360.0). For an Enumerated data entity called WORKING_DAYS inheriting from a concept data entity called WEEK_DAYS, the range of the enumeration values could be as follows: (Monday, Friday).
Attribute_Inheritance	: INHERITED.

3.2.4.4.4 TEXT_SIZE

Attribute_Name	: TEXT_SIZE
Attribute_Definition	: The limitation on the size of the values of a text. This attribute specifies the minimum and the maximum number of characters the text may contain. If the minimum and the maximum are equal, i.e. the exact size of the text is known, then a single value is provided.
Attribute_Obligation	: Conditional.
Attribute_Condition	: This attribute is mandatory if the DATA_TYPE is Text.
Attribute_Value_Type	: Integer or (Integer, Integer). The integer can be expressed as a literal value or a constant name.
Attribute_Maximum_Occurrence	: 1.
Attribute_Comment	: This attribute only applies for Text data entities.
Attribute_Inheritance	: INHERITED.

3.2.4.4.5 CONSTANT_VALUE

Attribute_Name	: CONSTANT_VALUE
Attribute_Definition	: The value of this attribute is the value given to a constant (entity whose CLASS attribute is set to CONSTANT).
Attribute_Obligation	: Optional.
Attribute_Condition	: This attribute is permitted if CLASS attribute is set to CONSTANT.
Attribute_Value_Type	: Entity_Type
Attribute_Maximum_Occurrence	: 1.
Attribute_Inheritance	: NOT_INHERITED.

3.2.5 USER-DEFINED ATTRIBUTES

The standard attributes specified in Section 3.2.4 are those predefined by this recommendation and must be recognized by any system that states conformance to this recommendation. It is recognized that there may be further attributes that are more specific to a particular domain, mission or project.

This section defines the mechanism used to define specific attributes; then these attributes can be used in the same manner as the standard ones to define the semantics of particular data entities. These attributes shall be called '**user-defined attributes**'.

The standard attributes shall be used prior to any other specific user-defined attributes. This means that, if a standard attribute already covers the need, a user-defined attribute cannot be defined for this need in a data entity definition.

Further, it is not allowed to define a specific attribute which has the name of a standard attribute.

3.2.5.1 MECHANISM FOR NEW ATTRIBUTE DEFINITIONS

A **User-Defined Attribute** is defined using the attribute descriptors specified in section 3.1. The description of a User-Defined Attribute follows the same rules than the one of a standard attribute.

The mandatory descriptors are :

- attribute_name,
- attribute_definition,
- attribute_obligation,
- attribute_value_type,
- attribute_maximum_occurrence,
- attribute_inheritance.

The conditional descriptors are :

- attribute_condition,
- attribute_enumeration_values,
- attribute_maximum_size.

The optional descriptors are :

- attribute_value_example,
- attribute_default_value,
- attribute_comment.

The following example gives the definition of the user-defined attribute **ASSOCIATED_UTILITIES** providing the name of the utility which processes the physical values of a specific data entity within a data product.

Attribute_Name	: ASSOCIATED_UTILITIES
Attribute_Definition	: Attribute that provides the name of the utility used to process the occurrences of the data entity with which the attribute is associated.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: (Text, Text)
Attribute_Comment	: The first text gives the name of the utility, and the second one specifies the application context of the utility.
Attribute_Inheritance	: INHERITED

This second example shows how to define an external reference.

Attribute_Name	: AUDIO_EXAMPLE
Attribute_Definition	: The AUDIO_EXAMPLE attribute is a pointer to a sound file in .wav format providing an example to be heard.
Attribute_Obligation	: Optional.
Attribute_Value_Type	: Text
Attribute_Comment	: The value is expressed as a path.
Attribute_Inheritance	: INHERITED

Example 3-4: Defining user-defined attribute

The attribute usage in a data entity definition would be as follows:

NAME	: MUSIC_INSTRUMENT
CLASS	: CONCEPT
DEFINITION	: "It corresponds to an instrument".
DATA_TYPE	: Enumerated
ENUMERATION_VALUES	: (Piano, Guitar, Violin, Saxophone)
AUDIO_EXAMPLE	: "/home/MUSIC/INSTRUMENTS/SOUND/Piano.wav"

Example 3-5: Usage of a user-defined attribute

3.2.5.2 REGISTRATION OF USER-DEFINED ATTRIBUTES

So as to obtain maximum reuse and hence interoperability across missions, projects and agencies, it is desirable that new user-defined attributes which are created by projects are submitted for central registration. This means that they can be reused by other projects, eventually leading to a uniform data entity dictionary across many missions and projects.

The advantage of sharing the data entity dictionary is that software can be developed to handle the entity data descriptions, which can then be reused by many other projects.

To register user-defined attributes, the data description registration capabilities detailed in the CCSDS Recommendations on Control Authorities (see References [4] and [E5]) should be followed.

When a user registers a user-defined attribute, the following information must be included:

- Identification of the user - this information shall be as defined in the registration of data descriptions with the CCSDS Control Authority (see References [4] and [E5]).
- A specification of the user-defined attribute using the attributes defined in Section 3.1.
- If software is available to support processing of the value of the user-defined attribute, this should be submitted with the definition.

3.3 RELATIONSHIP RULES

3.3.1 REFERENCE TO AN EXTERNAL DEDSL DICTIONARY

As seen previously (in attributes definition) some references can be made in a Data Entity Dictionary to other ones.

In a Data Entity Dictionary definition, the `EXTERNAL_DICTIONARY_REFERENCE` attribute can be used (1 to n times) to reference the DEDSL dictionaries whose concepts are reused in the current one.

When a concept is reused (as underlined by the use of the `INHERITS_FROM` attribute) the origin of this concept can be given by the second value of this attribute. If this second value is omitted, it means that the description of the reused concept is assumed to be local to the current DEDSL dictionary or, if it is not the case, that there is a default dictionary referenced through the `EXTERNAL_DICTIONARY_REFERENCE` attribute which must be unique in that case.

3.3.2 COMPOSITION RELATION

As seen previously a data entity may be composite, that is, it can be made up of a series of other data entities. This notion is rendered by the composition relation for which the `COMPONENT` attribute is defined.

When a data entity is considered as being composite, it is not obligatory to mention all its component data entities if there are not known yet (in particular, when the dictionary is undergoing a definition process). However, when the data entity dictionary corresponds to a physical data product, the components have to be defined.

3.3.3 INHERITANCE

A concept (defined with the CLASS attribute set to CONCEPT) can be reused in other data entity descriptions. The INHERITS_FROM attribute must then be used in the data entity descriptions which have to refer to the wished concept.

Once a reference has been made to a concept in an entity description, the current description will then inherit the values of all the attributes of the concept which have been defined as INHERITED or INHERITED_AND_ENRICHABLE.

Optional and conditional INHERITED attributes such as DATA_TYPE can be defined locally for the current description each time that they have not been defined for the concept description.

INHERITED_AND_ENRICHABLE attributes such as SPECIFIC_INSTANCE can be defined locally and their value enriches the information expressed at the concept level.

The attributes which have been defined as NOT_INHERITED such as NAME, CLASS, ... have to be defined locally.

A concept may be used to define other concepts. A concept can also be used to qualify data fields (physical bits) of a data product.

Example: a concept defined in a domain dictionary (e.g. gamma rays) can be specialized in a specific product description (e.g. specifying units) and used to qualify several measurements.

As seen in Section 3.1.10., the COMPONENT attribute is inherited. This attribute is also enrichable; consequently, components can be added in order to specialize the definition of the data entity with regards to the inherited concept.

3.3.4 OTHER KINDS OF RELATION

In addition to the relationships handled with a specific attribute (see above) some other ones have to be described in DEDSL dictionaries. In that case, there are two possible ways to proceed:

- Define a precise user-defined attribute with appropriate descriptors and use it¹,
- Use the generic standard relation attribute name "relation" and express the relationship in free text.

¹Remark: It may be interesting to define some specialized attributes to facilitate some research even if the relation is also expressed in free text.

Example: an attribute SIGNIFICANT_IF can point to a validity flag related to the current definition.

3.4 IMPLEMENTATION GUIDELINES

This section gives some guidelines concerning the use of attributes, whether 'standard', 'user-defined'.

- Although for any single semantic entity description, the standard attributes can be presented or accessed in any order, it is recommended that the following order should

be used whenever possible so as to present a common style to all users. The mandatory attributes are indicated in bold characters:

NAME
ALIAS
CLASS
DEFINITION
SHORT_DEFINITION
COMMENT
UNITS
SPECIFIC_INSTANCE
INHERITS_FROM
COMPONENT
KEYWORDS
RELATION
DATA_TYPE
ENUMERATION_VALUES
ENUMERATION_MEANING
ENUMERATION_CODING
RANGE
TEXT_SIZE
CONSTANT_VALUE

- User-defined attributes should follow the standard attributes.
- Although for any single user-defined attribute definition, the descriptors defining the user-defined attribute can be presented or accessed in any order, it is recommended that the following order should be used whenever possible so as to present a common style to all users. The mandatory descriptors are indicated in bold characters:

ATTRIBUTE_NAME
ATTRIBUTE_DEFINITION
ATTRIBUTE_OBLIGATION
ATTRIBUTE_CONDITION
ATTRIBUTE_MAXIMUM_OCCURRENCE
ATTRIBUTE_VALUE_TYPE
ATTRIBUTE_MAXIMUM_SIZE
ATTRIBUTE_ENUMERATION_VALUES
ATTRIBUTE_COMMENT
ATTRIBUTE_INHERITANCE
ATTRIBUTE_DEFAULT_VALUE
ATTRIBUTE_VALUE_EXAMPLE

- The attributes for each entity description (or descriptors for each attribute definition) must be grouped in some manner so as to keep them separate from the attributes of other entity descriptions. The methodology for grouping the attributes must be defined formally in the DEDSL implementation syntax.
- It is strongly recommended to define a maximum length for identifiers and texts for readability reasons and for a possible automatic processing of the DED.

4 DEDSL CONFORMANCE: ABSTRACT DEDSL (ADID = CCSD0011)

The assignment of unique identifiers, in the form of CCSDS ADID, is needed for the DEDSL material in order to facilitate the automated interpretation of data entity dictionaries conforming to the DEDSL.

This DEDSL specification (Section 3) defines a set of standard attributes by name with restrictions on their permitted values. Note that this part of the specification does not specify how the attribute names and values are to be linked to any given data object. This allows a variety of formatting approaches to be used for this linking and this is the first primary way of using the DEDSL.

As an example of usage, consider a data object (see References [1] and [E4] for detailed explanations of the concepts of data objects, LVO) with an associated registered description. The description could contain an LVO with Class ID = E (meaning semantic description data), which in turn, contains a set of attributes and values that apply to numerous instances of the data object type.

If the attributes in the LVO with Class ID = E conform to the DEDSL specification of Section 3, then it would carry the ADID assigned to Section 3 of this Recommendation which is **CCSD0011**.

5 ANNEX A: DEDSL EXAMPLES

In this section a community DED is presented showing the semantic information relative to the data entities chosen as being concepts. Then the definition of a product DED using this community DED for the definition of some of its data entities.

5.1 COMMUNITY DED

a) Dictionary attributes

DICTIONARY_NAME	: Planetary_Science_Data_Dictionary
DICTIONARY_DEFINITION	: "This dictionary contains data entity definitions relative to planetary science and which may be re-used for defining data products."
DICTIONARY_VERSION	: "1.0"
CASE_SENSITIVITY	: NOT_CASE_SENSITIVE
DEDSL_VERSION	: "2.2"
TEXT_FIELD_CHARACTER_SET	: ""

b) Dictionary entities

NAME	: LATITUDE
ALIAS	: (LAT, "Used by the historical projects EARTH_PLANET")
CLASS	: CONCEPT
DEFINITION	: "Latitudes north of the equator shall be designated by the use of the plus (+) sign, latitudes south of the equator shall be designated by the use of the minus sign (-). The equator shall be designated by the use of the plus sign (+)."
SHORT_DEFINITION	: "Latitude"
UNITS	: deg
SPECIFIC_INSTANCE	: (+00.000, "Equator")
DATA_TYPE	: REAL
RANGE	: (-90.00, +90.00)

NAME	: LONGITUDE
ALIAS	: (LON, "Used by the historical projects EARTH_PLANET")
CLASS	: CONCEPT
DEFINITION	: "Longitudes east of Greenwich shall be designated by the use of the plus sign (+), longitudes west of Greenwich shall be designated by the use of the minus sign (-). The Prime Meridian shall be designated by the use of the plus sign (+). The 180 th meridian shall be designated by the use of the minus sign (-)."
SHORT_DEFINITION	: "Longitude"
UNITS	: deg
SPECIFIC_INSTANCE	: (-180.000, "The 180 th Meridian")
DATA_TYPE	: REAL
RANGE	: (-180.00, +180.00)

NAME	: PRODUCT_ID
ALIAS	: (PRODUCT_NAME, "Used by the historical projects EARTH_PLANET to identify their data products")
CLASS	: CONCEPT
DEFINITION	: "The PRODUCT_ID represents a permanent, unique identifier assigned to a data product by its producer."
SHORT_DEFINITION	: "Product Identification"
DATA_TYPE	: TEXT
TEXT_SIZE	: 40

5.2 DICTIONARY ASSOCIATED WITH PRODUCT_X

The concepts of LATITUDE, LONGITUDE and PRODUCT_ID match the data entities appearing within the data product PRODUCT_X and therefore they are referenced within the current dictionary.

a) Dictionary attributes

DICTIONARY_NAME	: PRODUCT_X_Dictionary
DICTIONARY_DEFINITION	: "This dictionary contains the data entity definitions relative to the data product PRODUCT_X."
DICTIONARY_VERSION	: "1.0"
CASE_SENSITIVITY	: NOT_CASE_SENSITIVE
DEDSL_VERSION	: "2.3"
TEXT_FIELD_CHARACTER_SET	: ""
EXTERNAL_REFERENCE_DICTIONARY	: (Planetary_Science_Data_Dictionary)

b) Dictionary entities

NAME : HEADER
CLASS : DATA_FIELD
DEFINITION : "It represents the header of the data product PRODUCT_X. It identifies an aggregation of values which are associated with an image array."

SHORT_DEFINITION : "Image Header Values"
COMPONENT : PRODUCT_ID_X(1 .. 1)
COMPONENT : ACQ_STATION(1 .. 1)
COMPONENT : ACQ_TIME(1 .. 1)
COMPONENT : CENTRE_COORD(1 .. 1)
DATA_TYPE : COMPOSITE

NAME : PRODUCT_ID_X
CLASS : DATA_FIELD
DEFINITION : "It represents a permanent, unique identifier assigned to the data product PRODUCT_X."

SHORT_DEFINITION : "Product Identification"
INHERITS_FROM : (PRODUCT_ID)

NAME : ACQ_STATION
ALIAS : (ACQUSTAT, "used in the FITS header")
CLASS : DATA_FIELD
DEFINITION : "It includes the identifier of the station which has acquired the data."

SHORT_DEFINITION : "Identifier of the acquisition station"
DATA_TYPE : Enumerated
ENUMERATION_VALUES : (AMERICA, EUROPE, ASIA)
ENUMERATION_MEANING : "(AMERICA : station located in America, EUROPE : station located in Europe, ASIA : station located in Asia)"

NAME : ACQ_TIME
ALIAS : (ACQUTIME, "Used in the FITS header")
CLASS : DATA_FIELD
DEFINITION : "It represents the date and time of the acquisition of the data. Its format is the following one: YYYY-MM-DDThh:mm:ss.d_>Z. It conforms to the CCSDS ISO rules for date/time definitions. The acquisition time should correspond to the first scan line of the data."

SHORT_DEFINITION : "Date/Time of the data acquisition"
DATA_TYPE : Text
TEXT_SIZE : 40

NAME : CENTRE_COORD
CLASS : DATA_FIELD
DEFINITION : "It represents a coordinate centre."
SHORT_DEFINITION : "Centre coordinates"
COMPONENT : LATITUDE_X(1 .. 1)
COMPONENT : LONGITUDE_X(1 .. 1)
KEYWORDS : "LATITUDE BY LONGITUDE COORDINATE CENTRE"
DATA_TYPE : COMPOSITE

NAME : LATITUDE_X
CLASS : DATA_FIELD
DEFINITION : "It represents the latitude used for the center coordinate."
INHERITS_FROM : (LATITUDE)

NAME : LONGITUDE_X
CLASS : DATA_FIELD
DEFINITION : "It represents the longitude used for the center coordinate."
SHORT_DEFINITION : "Product Identification"
INHERITS_FROM : (LONGITUDE)

NAME : W_IMAGE_SIZE
CLASS : CONSTANT
DEFINITION : "It represents the number of pixels for an image taken from spacecraft W."
SHORT_DEFINITION : "Spacecraft W Image pixel"
RELATION : ("number of pixels of an spacecraft W image", DATA_1)
DATA_TYPE : INTEGER
CONSTANT_VALUE : 1 440 000

NAME : DATA_1
CLASS : DATA_FIELD
DEFINITION : "It represents an image taken from spacecraft W."
SHORT_DEFINITION : "Spacecraft W Image"
COMMENT : "The image is an array of W_IMAGE_SIZE items called DATA_1_PIXEL"
COMPONENT : DATA_1_PIXEL (1 .. W_IMAGE_SIZE)
KEYWORDS : "IMAGE"
DATA_TYPE : COMPOSITE

DRAFT RECOMMENDATION FOR DEDSL

NAME	: DATA_1_PIXEL
CLASS	: DATA_FIELD
DEFINITION	: "It represents a pixel belonging to the image taken from spacecraft W."
SHORT_DEFINITION	: "Spacecraft W Image pixel"
DATA_TYPE	: INTEGER
RANGE	: (0 , 255)

6 ANNEX B: INFORMATIVE REFERENCES

(This annex **is not** part of the Recommendation)

This annex provides a list of references which may be valuable to the user of this Recommendation as background material or to provide implementation guidelines for using this Recommendation.

- [E1] *Procedures Manual for the Consultative Committee for Space Data Systems*, CCSDS A00.0-Y-6. Yellow Book. Issue 6. Washington, D.C.: CCSDS, May 1994.
- [E2] *The Data Description Language EAST - Specification (CCSD0010)*. Recommendation for Space Data System Standards, CCSDS 644.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, May 1997.
- [E3] *The Data Description Language EAST - A Tutorial*. Report Concerning Space Data System Standards, CCSDS 645.0-G-1. Green Book. Issue 1. Washington, D.C.: CCSDS, May 1997.
- [E4] *Standard Formatted Data Units - A Tutorial*. Report Concerning Space Data System Standards, CCSDS 621.0-G-1. Green Book. Issue 1. Washington, D.C.: CCSDS, May 1992.
- [E5] *Standard Formatted Data Units - Control Authority Procedures Tutorial*. Report Concerning Space Data System Standards, CCSDS 631.0-G-2. Green Book. Issue 2. Washington, D.C.: CCSDS, November 1994.
- [E6] *Hierarchical Data Format (HDF)*. Version 4.0r. National Centre for Supercomputing Applications (NCSA).
- [E7] *Common Data Format (CDF)*. Version 2.5.19a. National Space Science Data Center, May 17 1996.
- [E8] *Time Code Formats*. Recommendation for Space Data System Standards, CCSDS 301.0-B-2. Blue Book. Issue 2. CCSDS April 1990.
- [E9] *UNIDATA Units Package*. NCAR, Version 1.11.5, 18 August 1997.
- [E10] *Information Processing - Representation of SI and other units in systems with limited character sets*. ISO 2955-1983. Geneva:ISO,1983.